

Query/C mmand : prt fu

1 / 1 JAPIO - ©JPO - image

PN - JP 06110715 A 19940422 [JP06110715]
TI - DYNAMIC ALLOCATING METHOD FOR COMPUTER RESOURCES IN
VIRTUAL COMPUTER SYSTEM
IN - TANAKA TOSHIHARU; ARAI TOSHIAKI; SUGAMA NOBUYOSHI; YAMAMOTO
TADASHI; IKEGAYA NAOKO
PA - HITACHI LTD
AP - JP25701392 19920925 [1992JP-0257013]
IC1 - G06F-009/46
AB - **PURPOSE:** To provide a virtual computer system which is dynamically changing the
allocation of a main storage area to an OS(operating system).
CONSTITUTION: With the occurrence of a specific event as an opportunity, a high-
order address area within the guest main storage area 420-2 of a virtual computer VM-2
is made to an OFF line (unconnection state). Then, a virtual computer monitor 410
moves the guest main storage area 420-2 of the VM-2 and allows the high-order address
area which is adjacent to the guest main storage area 420-1 of a virtual computer VM-1
to be an empty state. Subsequently, the high-order address area which is adjacent to the
guest main storage area 420-1 of the VM-1 is made in ON-line (connection state) and is
connected with the VM-1. Thus, the allocation of a main storage area to an OS is
dynamically changed.
COPYRIGHT: (C)1994,JPO&Japio

Search statement 3

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平6-110715

(43)公開日 平成6年(1994)4月22日

(51)Int.Cl.⁵

G 0 6 F 9/46

識別記号

3 5 0

庁内整理番号

8120-5B

F I

技術表示箇所

審査請求 未請求 請求項の数9(全14頁)

(21)出願番号 特願平4-257013

(22)出願日 平成4年(1992)9月25日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 田中 俊治

神奈川県川崎市麻生区王禅寺1099 株式会
社日立製作所システム開発研究所内

(72)発明者 新井 利明

神奈川県川崎市麻生区王禅寺1099 株式会
社日立製作所システム開発研究所内

(72)発明者 須釜 延芳

神奈川県横浜市戸塚区戸塚町5030番地 株
式会社日立製作所ソフトウェア開発本部内

(74)代理人 弁理士 富田 和子

最終頁に続く

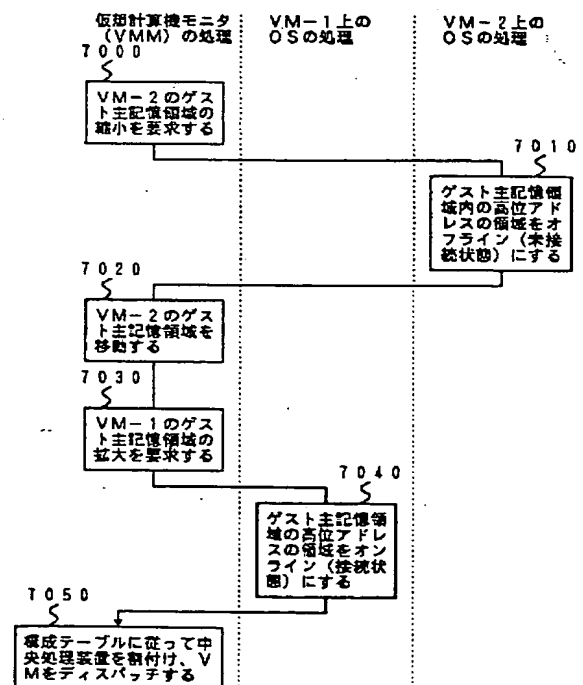
(54)【発明の名称】 仮想計算機システムにおける計算機資源の動的割付け方法

(57)【要約】

【目的】 OSへの主記憶領域の割付けを動的に変更可能な仮想計算機システムを提供する。

【構成】 特定のイベントの発生を契機として、仮想計算機VM-2のゲスト主記憶領域内の高位アドレスの領域をオフライン(未接続状態)にする(ステップ7010)。次に、仮想計算機モニタ410は、VM-2のゲスト主記憶領域を移動(ステップ7020)して、仮想計算機VM-1のゲスト主記憶領域に隣接する高位アドレスの領域を空き状態とする。その後、VM-1のゲスト主記憶領域に隣接する高位アドレスの領域をオンライン(接続状態)にして、VM-1に接続する(ステップ7040)。このようにして、OSへの主記憶領域の割付けを動的に変更する。

計算機資源の割り当て変更方法のフローチャート(図1)



【特許請求の範囲】

【請求項1】中央処理装置と、主記憶装置と、前記中央処理装置からの指示に応じて当該中央処理装置と非同期に処理を行う非同期処理装置とを含む計算機上で複数のオペレーティング・システム（OS）が仮想計算機モニタの制御下で走行する仮想計算機システムにおいて、前記複数のOSのうち、第1のOSに対して、前記主記憶装置の第1の固定領域を割付けるとともに、第2のOSに対して前記第1の固定領域のアドレス高位側に隣接する第2の固定領域を割り付けるステップと、特定のイベントが発生した時に、前記第2のOSに割り当てられた第2の固定領域のサイズを縮小するステップと、該サイズを縮小した第2の固定領域を前記主記憶装置上でアドレス高位側へ移動することにより、前記第1の固定領域のアドレス高位側に空き領域を生成するステップと、前記第1のOSの主記憶領域のサイズを拡大するステップとを有する、仮想計算機システムにおける計算機資源の割付け方法。

【請求項2】請求項1において、前記特定のイベントの発生前と発生後における各々のOSの主記憶領域のサイズを定義するステップと、前記特定のイベントが発生したとき、前記定義に従って前記複数のOSの主記憶領域のサイズを変更するステップとを有する、仮想計算機システムにおける計算機資源の動的割付け方法。

【請求項3】請求項1において、前記中央処理装置を複数個有し、特定のイベントの発生前と発生後において各々のOSに割り付ける中央処理装置を定義するステップと、前記特定のイベントが発生したとき、前記定義に従って前記複数のOSに中央処理装置を割り付けるステップとを有する、仮想計算機システムにおける計算機資源の動的割付け方法。

【請求項4】請求項1、2、または3において、前記特定のイベントは、特定の時間になったというイベントであることを特徴とする、仮想計算機システムにおける計算機資源の動的割付け方法。

【請求項5】請求項1、2、または3において、前記特定のイベントは、前記計算機と異なる第2の計算機に障害が発生し、該第2の計算機の処理を前記第1のOSが引き継いだというイベントであることを特徴とする、仮想計算機システムにおける計算機資源の動的割付け方法。

【請求項6】請求項1の前記空き領域を生成するステップは、前記第2のOSの主記憶領域を前記仮想計算機モニタが生成した仮想空間とし、該仮想空間を前記縮小された第2の固定領域にマッピングするステップと、前記仮想空間のマッピング先を前記主記憶装置の移動先の領域に変更するステップと、該移動先の領域を前記第2のOSの固定領域として割り付けるステップとを有するこ

とを特徴とする、仮想計算機システムにおける計算機資源の動的割付け方法。

【請求項7】請求項6において、前記第2のOSの主記憶領域の移動中は、前記第2のOSが発行した前記非同期処理装置の使用開始命令を、前記仮想計算機モニタがシミュレーションするステップと、該シミュレーション時に前記非同期処理装置が前記第2のOSの主記憶領域のうちどの領域をアクセスするかを解析するステップと、該アクセス対象となる領域は、前記非同期処理装置によるアクセス終了後、移動するステップとを有することを特徴とする、仮想計算機システムにおける計算機資源の動的割付け方法。

【請求項8】中央処理装置と、主記憶装置と、前記中央処理装置からの指示に応じて当該中央処理装置と非同期に処理を行う非同期処理装置とを含む計算機上で複数のオペレーティング・システム（OS）が仮想計算機モニタの制御下で走行する仮想計算機システムにおいて、前記いずれかのOSの主記憶領域として割り付けられた前記主記憶装置の連続アドレスの固定領域を移動するとき、当該OSの主記憶領域と前記固定領域との対応関係を前記仮想計算機モニタが生成した仮想空間に定義するステップと、前記仮想空間のマッピング先を前記主記憶装置上の移動先の連続アドレスの固定領域に変更するステップと、該移動先の固定領域を前記OSの主記憶領域として割り付けるステップとを有する、仮想計算機システムにおける計算機資源の動的割付け方法。

【請求項9】請求項8において、前記定義するステップの後に、前記仮想空間のマッピング先の固定領域のサイズを前記OSの主記憶領域のサイズよりも小さくするステップを有することを特徴とする、仮想計算機システムにおける計算機資源の動的割付け方法。

【発明の詳細な説明】**【0001】**

【産業上の利用分野】本発明は、計算機システムに関し、特に、1台の計算機上で複数のOSが走行する計算機システムにおいて、OSへの計算機資源の割付けを動的に変更するのに好適な計算機システムに関する。

【0002】

【従来の技術】1台の計算機上で複数のオペレーティング・システム（OS：Operating System）が走行可能な計算機システムとして、仮想計算機システム（VMS：Virtual Machine System）がある。仮想計算機システムは、1台の計算機上に論理的な計算機である仮想計算機（VM：Virtual Machine）を複数台生成し、各仮想計算機（VM）において対応する1つのOSの走行を可能とする。この仮想計算機システムは、使用目的の異なる複数のOSを1台の計算機上で運用したり、旧バージョンのOSを運用しながら新バージョンのOSのシステムゼネレーションを行ったり、1台の計算機上でOSの

テストを並列に実行するために主に利用されている。仮想計算機システムにおいて、その制御プログラムである仮想計算機モニタ (VMM: Virtual Machine Monitor) は、各仮想計算機 (VM) に計算機資源を分配し、仮想計算機 (VM) のスケジューリングとディスパッチを行なう。また、仮想計算機モニタ (VMM) は、仮想計算機上のOSが発行した命令のうち計算機では直接実行できない命令、すなわち、ソフトウェアの介入を必要とする命令のシミュレーション処理等を行なう。

【0003】仮想計算機には、仮想計算機の主記憶領域 (ゲスト主記憶領域) の与え方によってページ可能VMとページ固定VMがある。ページ可能VMは、特開昭57-212680号公報記載のように、仮想計算機モニタ (VMM) が生成した仮想空間 (ホスト仮想空間) 上に、ゲスト主記憶領域を常駐させる。従って、ページ可能VMのゲスト主記憶領域は、仮想計算機モニタ (VMM) によるページングの対象となる。このページングにより、ページ可能VMのゲスト主記憶領域のサイズは、ホスト仮想空間のマッピング先である仮想計算機モニタ (VMM) のページング領域のサイズよりも大きくできる。

【0004】他方、ページ固定VMは、特開昭60-24735号公報記載のように、主記憶装置上の連続領域にゲスト主記憶領域を常駐させる。従って、ページ固定VMのゲスト主記憶領域は、仮想計算機モニタ (VMM) によるページングの対象とならない。このため、ページ固定VMは、ページ可能VMに比べてアドレス変換に要するオーバーヘッドが少ない。しかし、ページ固定VMのゲスト主記憶領域のサイズは、対応付けられた主記憶装置の連続領域の大きさしか設定できない。

【0005】また、特開平3-211629号公報は、中央処理装置とは非同期に処理を行なうチャネル等の非同期処理装置によるゲスト主記憶領域へのアクセスを一時停止して、主記憶装置上でページ固定VMのゲスト主記憶領域を移動可能とする仮想計算機システムを開示している。この仮想計算機システムによれば、主記憶装置の記憶領域の断片化を、ゲスト主記憶領域を移動することにより防止できる。

【0006】さらに、特開平2-201655号公報は、1台の計算機を構成する複数の中央処理装置を、仮想計算機の中央処理装置として割り付ける仮想計算機割り当て方式を開示している。

【0007】

【発明が解決しようとする課題】仮想計算機システムにおけるゲスト主記憶領域の割り当てには、以下の課題がある。

【0008】(1) 第1の課題は、ゲスト主記憶領域のサイズのトータルバランスに関する。

【0009】仮想計算機システムの運用では、仮想計算機の負荷変動が定期的に発生することが多い。例えば、

2台の仮想計算機が終日運用される場合、第1の仮想計算機では、TSS (Time Sharing System) を運用しているために昼間の負荷が高く、第2の仮想計算機では、空き時間にバッチジョブを実行するために夜の負荷が高い、という使い方も多い。このような場合、負荷の高い方の仮想計算機に、より多くの主記憶領域を与えた方が、ページング処理等を削減でき、全体のスループットが向上する。

【0010】また、図1に示すように第1の計算機500に対して第2の計算機600上の仮想計算機VM-1がホットスタンバイ状態にある場合、第1の計算機500が正常に動作している間は、仮想計算機VM-1には、比較的少量のゲスト主記憶領域420-1を与え、バッチジョブやTSSを運用する仮想計算機VM-2のゲスト主記憶領域420-2を大きくしておいた方がよい。これは、仮想計算機VM-1では、第1の計算機500に障害が発生した場合に備えた処理のみを行なっているからである。第1の計算機500の障害によりシステム切り替え装置700が、第1の計算機500の処理を仮想計算機VM-1に切り替えた場合、この仮想計算機VM-1のゲスト主記憶領域420-1のサイズを大きくして、高い負荷に耐えられるようにすることが望ましい。

【0011】しかしながら、従来、仮想計算機のゲスト主記憶領域のサイズを変更するには、他の仮想計算機の主記憶領域の配置も変更しなければならないため、全ての仮想計算機を停止しなければならなかった。

【0012】(2) 第2の課題は、ゲスト主記憶領域の主記憶装置上での移動に関する。仮想計算機上のOSが入出力命令で実行を要求するチャネルプログラム (ゲストチャネルプログラム) は、ゲスト主記憶領域上の絶対アドレス (ゲスト絶対アドレス) で記述されている。このため、OS発行の入出力命令の仮想計算機モニタ (VMM) によるシミュレーションでは、データアドレスを主記憶装置での絶対アドレス (ホスト絶対アドレス) に変換したチャネルプログラム (ホストチャネルプログラム) を作成し、次に、このホストチャネルプログラムを実行する。従って、ゲスト主記憶領域の移動前に実行を開始したホストチャネルプログラムのデータアドレスは、全て、移動前のゲスト主記憶領域を示している。このため、特開平3-211629号公報が開示する方法によって、ゲスト主記憶領域を移動後、仮想計算機モニタによりホストチャネルプログラムの実行を再開すると、移動前の領域のデータのみが更新され、移動後のゲスト主記憶領域のデータは更新されない。従って、データの不一致が発生してしまう。

【0013】(3) 第3の課題は、ページ固定VMからページ可能VMへの変更に関する。新たな仮想計算機の生成等により主記憶領域が必要な場合、ページ固定VMをページ可能VMとして、空き領域を作ることが望まし

い。この際、ゲスト主記憶領域のサイズ変更、移動等を伴うので、上記と同様の課題がある。

【0014】(4)第4の課題は、ページ可能VMからページ固定VMへの変更に関する。主記憶装置上に十分な空き領域がある場合、性能を向上させるためにページ可能VMをページ固定VMとすることが望ましい。この場合にも、通常、ゲスト主記憶領域のサイズ変更、移動等を伴うので上記と同様の課題がある。

【0015】(5)第5の課題は、仮想計算機への中央処理装置の割り当てに関する。仮想計算機の負荷変動が定期的に発生する場合や、ホットスタンバイ状態であった仮想計算機にシステムが切り替わった場合、高い負荷の仮想計算機に対して優先的に、中央処理装置を与えるべきである。しかしながら、特開平2-201655号公報が示すように、従来、仮想計算機への中央処理装置の割り当ては、仮想計算機の生成時に固定的に決められていた。

【0016】本発明の目的は、各仮想計算機の負荷変動に応じた計算機資源の再配分を可能とする、仮想計算機システムにおける計算機資源の動的割付け方法を提供することにある。

【0017】本発明の他の目的は、ゲスト主記憶領域のサイズを動的に変更可能な仮想計算機システムを提供することにある。

【0018】本発明のさらに他の目的は、チャネル等の非同期処理装置によるゲスト主記憶領域へのアクセスを停止することなく、ゲスト主記憶領域を主記憶装置上で移動可能な、仮想計算機システムにおける計算機資源の動的割付け方法を提供することにある。

【0019】また、本発明の別の目的は、仮想計算機への中央処理装置の割り当てを動的に変更可能な、仮想計算機システムにおける計算機資源の動的割付け方法を提供することにある。

【0020】

【課題を解決するための手段】本発明による仮想計算機システムにおける計算機資源の割付け方法は、中央処理装置と、主記憶装置と、前記中央処理装置からの指示に応じて当該中央処理装置と非同期に処理を行う非同期処理装置とを含む計算機上で複数のオペレーティング・システム(OS)が仮想計算機モニタの制御下で走行する仮想計算機システムにおいて、前記複数のOSのうち、第1のOSに対して、前記主記憶装置の第1の固定領域を割付けるとともに、第2のOSに対して前記第1の固定領域のアドレス高位側に隣接する第2の固定領域を割り付けるステップと、特定のイベントが発生した時に、前記第2のOSに割り当てられた第2の固定領域のサイズを縮小するステップと、該サイズを縮小した第2の固定領域を前記主記憶装置上でアドレス高位側へ移動することにより、前記第1の固定領域のアドレス高位側に空き領域を生成するステップと、前記第1のOSの主記憶

領域のサイズを拡大するステップとを有するものである。

【0021】前記空き領域を生成するステップは、好ましくは、前記第2のOSの主記憶領域を前記仮想計算機モニタが生成した仮想空間とし、該仮想空間を前記縮小された第2の固定領域にマッピングするステップと、前記仮想空間のマッピング先を前記主記憶装置の移動先の領域に変更するステップと、該移動先の領域を前記第2のOSの固定領域として割り付けるステップとを有する。

【0022】また、前記第2のOSの主記憶領域の移動中は、前記第2のOSが発行した前記非同期処理装置の使用開始命令を、前記仮想計算機モニタがシミュレーションするステップと、該シミュレーション時に前記非同期処理装置が前記第2のOSの主記憶領域のうちどの領域をアクセスするかを解析するステップと、該アクセス対象となる領域は、前記非同期処理装置によるアクセス終了後、移動するステップとを有することが望ましい。

【0023】

【作用】

(1)各仮想計算機の生成時等に、特定のイベントの発生前と発生後における、この仮想計算機の計算機資源を構成テーブルに定義しておく。そして、特定のイベントが発生した時に、第2のOS(仮想計算機)の主記憶領域(ゲスト主記憶領域)のサイズを構成テーブルの定義に従って縮小し、各仮想計算機のゲスト主記憶領域を主記憶装置上で移動することにより、第1の仮想計算機のゲスト主記憶領域に隣接するアドレス高位の領域を空き領域にする。その後、第1の仮想計算機のゲスト主記憶領域のサイズを構成テーブルの定義に従って拡大する。

【0024】また、このとき構成テーブルの定義に従って仮想計算機への中央処理装置の割り当てを変更することもできる。たとえば、3台の仮想計算機が2台の中央処理装置を共用していた状態から、第1と第2の仮想計算機が第1の中央処理装置を共用し、第1の仮想計算機が第2の中央処理装置を専有する状態に遷移する。

【0025】ここで、特定のイベントとは、特定の時間になったというイベントや、仮想計算機が走行する計算機と異なる第2の計算機に障害が発生し、この第2の計算機の処理を第1の仮想計算機上のOSが引き継いだというイベントを示す。

【0026】以上のようにして、ゲスト主記憶領域のサイズを動的に変更可能な仮想計算機システムを提供できる。

【0027】また、仮想計算機への中央処理装置の割り当てを動的に変更可能な仮想計算機システムを提供できる。

【0028】(2)ページ固定VMのゲスト主記憶領域を、主記憶装置上で移動するとき、ゲスト主記憶領域と主記憶装置上の連続領域との対応関係を仮想計算機モニタ(VMM)が生成した仮想空間(ホスト仮想空間)に

定義し、ホスト仮想空間のマッピング先を移動先領域に変更し、移動後の主記憶装置上の連続領域をOSのゲスト主記憶領域として割り付ける。また、ゲスト主記憶領域の移動中は、OSによる非同期処理装置の使用開始命令は、仮想計算機モニタ(VMM)がシミュレーションし、非同期処理装置によるゲスト主記憶領域のアクセス対象領域は、そのアクセス完了後に、移動する。この際、ページ可能VMの利用により、ゲスト主記憶領域の移動中には、ホスト仮想空間に対応する主記憶領域はページ単位に不連続となることが許容される。このようにして、非同期処理装置によるゲスト主記憶領域のアクセス対象外の領域から順次、領域を移動する。このため、領域移動後に、非同期処理装置が誤って移動前の領域にアクセスすることはない。

【0029】以上のようにして、チャネル等の非同期処理装置によるゲスト主記憶領域へのアクセスを停止することなく、ゲスト主記憶領域を主記憶装置上で移動させることができる。

【0030】(3) ページ固定VMのゲスト主記憶領域とこれが常駐する主記憶装置上の連続領域の対応関係をホスト仮想空間に定義し、ホスト仮想空間の定義情報をアドレス変換に使用してOSを走行させる。また、ホスト仮想空間のマッピング先の領域をページアウトすることにより、ホスト仮想空間のマッピング先の領域サイズをゲスト主記憶領域サイズよりも小さくする。このようにして、動的にページ固定VMをページ可能VMに変更できる。

【0031】(4) ページ可能VMのゲスト主記憶領域と主記憶装置との対応関係を保持するホスト仮想空間のマッピング先の領域がページアウトされている場合、この領域を主記憶装置上にページインする。次に、ホスト仮想空間のマッピング先として、主記憶装置の連続領域を対応付ける。そして、この連続領域の主記憶装置における配置情報を使用してOSを走行させる。このようにして、動的にページ可能VMをページ固定VMに変更できる。

【0032】

【実施例】以下、本発明のゲスト主記憶領域の移動方法に関する第1の実施例と、計算機資源の割り付け変更方法に関する第2の実施例を、図を用いて詳細に説明する。

【0033】1. ゲスト主記憶領域の移動方法

図2に、本実施例の仮想計算機システム(VMS)の構成図を示す。図2において、100-1および100-2は中央処理装置、200は中央処理装置100-k($k=1, 2$)とは非同期的に処理を行なう非同期処理装置、300は主記憶装置400に対する参照及び更新を制御する記憶制御装置である。

【0034】非同期処理装置200は、計算機が通常の汎用大形計算機の場合には、チャネルである。チャネル

は、中央処理装置100-k($k=1, 2$)の動作と非同期に、入出力装置と主記憶装置400の間でデータ転送を行なう。また、計算機がスーパーコンピュータの場合には、非同期処理装置200は、ベクトルプロセッサである。ベクトルプロセッサは、中央処理装置100-k($k=1, 2$)の動作と非同期に、主記憶装置400上の命令列(いわゆる、ベクトル命令列)を実行する。本実施例では、初期状態として3台のページ固定VMが走行しており、仮想計算機モニタ(VMM)410の領域を除く主記憶装置400の記憶領域は、仮想計算機VM-1のゲスト主記憶領域420-1、仮想計算機VM-2のゲスト主記憶領域420-2、仮想計算機VM-3のゲスト主記憶領域420-3として各仮想計算機に割り付けられているものとする。

【0035】次に、図3を用いて仮想計算機モニタ410が、各仮想計算機の走行を制御するために作成する制御ブロック430-i($i=1, 2, 3$)について説明する。制御ブロック430-iは、各仮想計算機VM-i($i=1, 2, 3$)に対応して存在し、仮想計算機VM-i上のOSが中央処理装置100-k($k=1, 2$)において走行中の各種制御情報を与える。

【0036】図3において、直接実行ビット431が1のとき、仮想計算機上のOSが発行した非同期処理装置200の起動開始命令は直接実行され、0のとき、命令の実行は抑止されて、仮想計算機モニタ410を介して実行される。ページ固定VM識別ビット432が1であるとき仮想計算機は、ページ固定VMであることを示し、0のときページ可能VMであることを示す。ページ固定VM識別ビット432が1であるとき、起点アドレスフィールド433および終点アドレスフィールド434は、ゲスト主記憶領域の主記憶装置400における起点アドレスおよび終点アドレスを保持する。また、ページ固定VM識別子435が0であるとき、ホスト第1制御レジスタ435は、この仮想計算機のゲスト主記憶領域が常駐するホスト仮想空間を定義するホストセグメントテーブル(ホストST)の先頭アドレスを保持する。

【0037】次に、図4を用いて仮想計算機システム(VMS)におけるアドレス変換について説明する。

【0038】仮想計算機VM-2上のOSが生成した仮想空間(ゲスト仮想空間)上の論理アドレス(ゲスト論理アドレス)4000は、OSがゲスト主記憶領域420-2上に作成したゲストセグメントテーブル(ゲストST)およびゲストページテーブル(ゲストPT)によるアドレス変換4100の結果、ゲスト絶対アドレス4200に変換される。そして、仮想計算機がページ固定VMのとき、即ち、この仮想計算機の制御ブロック430のページ固定VM識別ビット432が1のとき、ページ固定VM用アドレス変換回路4300による変換結果がセレクト4500により有効化され、ホスト絶対アドレス4600となる。一方、仮想計算機がページ可能V

Mのとき、即ち、この仮想計算機の制御ブロック430のページ固定VM識別ビット432が0のとき、ページ可能VM用アドレス変換回路4400による変換結果がセクタ4500により有効化され、ホスト絶対アドレス4600となる。

【0039】ここで、ページ固定VM用アドレス変換回路4300によるゲスト絶対アドレス4200からホスト絶対アドレス4600への変換は、前記特開昭60-24735号公報に開示されているように、起点アドレスフィールド433の値を加算器4310によりゲスト絶対アドレス4200に加算することにより行なわれる。ただし、加算の結果の値が、終点アドレスフィールド434の値を超える場合、比較器4320は、プログラム割込みをOSに報告するために割込み信号4330を発生させる。

【0040】一方、ページ可能VM用アドレス変換回路4400によるゲスト絶対アドレス4200からホスト絶対アドレス4600への変換は、前記特開昭57-212680号公報に開示されているように、次のように行なわれる。まず、ゲスト絶対アドレス4200は、仮想計算機モニタ410からみればホスト仮想アドレスである。このため、ホスト第1制御レジスタ435の値とホスト仮想アドレス4410のセグメントフィールド(S)の値を加算器4420により加算してホストセグメントテーブル(ホストST)4430のエントリアドレスを得る。このエントリのページテーブルアドレスとホスト仮想アドレス4410のページフィールド(P)の値を加算器4440により加算してホストページテーブル(ホストPT)4450のエントリアドレスを得る。さらにこのエントリのページアドレスとホスト仮想アドレス4410のディスプレイメント(D)を結合することにより、ホスト絶対アドレス4600を得る。

【0041】次に、図5および図6を用いて仮想計算機の主記憶領域の主記憶装置400上での移動方法を説明する。図5はゲスト主記憶領域の移動方法のフローチャートであり、図6はゲスト主記憶領域の移動方法の説明図である。

【0042】図6(a)に示す初期状態において仮想計算機VM-3が走行を終了し、図6(b)に示すように、主記憶装置400の一部の領域が空き状態になったとする。そこで、以下に述べる方法により、仮想計算機VM-2のゲスト主記憶領域420-2を主記憶装置400上で移動する。

【0043】まず、仮想計算機モニタ410は、仮想計算機VM-2上のOSが発行した非同期処理装置200の起動開始命令をシミュレーションするために、仮想計算機VM-2の制御ブロック430-2内の直接実行ビット431を0に設定する(ステップ5000)。これにより、ゲスト主記憶領域420-2の移動中に、仮想計算機VM-2上のOSが非同期処理装置200の起動

開始命令を発行すると、命令の実行は抑止されて、仮想計算機モニタ410に割り込まれる。

【0044】次に、仮想計算機モニタ410は、ページ固定VMである仮想計算機VM-2に対して、図6

(c)に示すように、ホスト仮想空間450を生成し、VMをページ可能VMに変更する(ステップ5010)。このホスト仮想空間450を定義するホストセグメントテーブル(ホストST: Host Segment Table)およびホストページテーブル(ホストPT: Host Page Table)は、ホスト仮想アドレスに図6(c)に示すページング領域の起点アドレスを加算した値が、仮想計算機VM-2のゲスト主記憶領域のホスト絶対アドレスとなるようにテーブル値を設定し、仮想計算機モニタ410内の領域に作成する。即ち、ホスト仮想空間450の第jページが図6(c)に示すページング領域の第jページに対応付けられるように、仮想計算機モニタ410は、ホストSTおよびホストPTを作成する。

【0045】次に、仮想計算機モニタ410は、マッピング先の領域サイズを小さくする必要があるかを判定する(ステップ5020)。この判定は、仮想計算機システム(VMS)のユーザのコマンド入力等により、マッピング先の領域サイズを小さくするよう指定があったか否かにより判定する。指定があった場合、マッピング先の領域サイズを指定されたサイズに変更し、残りのフィールドはページアウトする。これによりゲスト主記憶領域のサイズよりもホスト仮想空間450のマッピング先の領域サイズの方を小さくする(ステップ5030)。ただし、非同期処理装置200によるアクセス対象のゲスト主記憶領域は、ページアウトの対象外とする。

【0046】次に、走行させるOSが、非同期処理装置200を使用中か否かを判定する(5035)。使用中でなければ、当該OSのホスト仮想空間の全領域のマッピング先を移動先領域に変更して(5040)、ステップ5100へ移行する。この移動先領域の変更は、移動前の領域の内容を移動後の領域にコピーし、ホストSTおよびホストPTの変更により、移動前の領域にマッピングされていたホスト仮想空間450上の領域を移動後の領域にマッピングすることにより行なう。ステップ5035で使用中ならば、図3の仮想計算機VM-2の制御ブロック430-2のページ固定VM識別ビット432に0を設定し、ホスト第1制御レジスタフィールド435にホストセグメントテーブル(ホストST)のアドレスを設定することによりホスト仮想空間450の定義情報を指定して仮想計算機VM-2上のOSをディスパッチする(ステップ5050)。この後、OSの走行(ステップ5060)が開始される。このOSが走行中は、仮想計算機VM-2の制御ブロック430-2内のページ固定VM識別ビット432は0なので、ゲスト絶対アドレスからホスト絶対アドレスへの変換は、仮想計算機VM-2の制御ブロック430-2のホスト第1制

御レジスタフィールド435に指定されたホストセグメントテーブル（ホストST）と、このホストSTが指定するホストページテーブル（ホストPT）に従って、行なわれる。

【0047】OS走行開始後、仮想計算機VM-2上のOSが非同期処理装置200の起動開始命令を発行（ステップ5070）すると、VM-2の制御ブロック430-2の直接実行ビット431は0なので、この命令は、仮想計算機モニタ410に割り出される。この結果、仮想計算機モニタ410は、非同期処理装置200の起動開始命令のシミュレーションを行なう（ステップ5080）。このシミュレーションでは、仮想空間の現在のマッピング状態に従ってデータ転送の対象とするアドレスの変換を行い、非同期処理装置200に起動開始命令を改めて発行する。起動が成功すれば（5083）、ステップ5085へ進み、未だ非同期処理装置使用中等の理由により起動に失敗すれば、ステップ5050へ戻る。

【0048】その後、図6（d）に示すように、非同期処理装置200のデータ転送に関係のないホスト仮想空間450のマッピング先を移動先領域に変更する（ステップ5085）。非同期処理装置200のデータ転送に関係のあるゲスト主記憶領域420-2内の領域は、非同期処理装置200の起動開始命令に指定されている。このため、該当するページ以外のマッピング先を、徐々に、移動先の領域に変更していく。

【0049】このシミュレーション後、仮想計算機モニタ410は、仮想計算機VM-2のゲスト主記憶領域420-2の移動を完了したかを判定する（ステップ5090）。この判定は、ホスト仮想空間450のマッピング先が、全て移動先の領域となったか否かで判定できる。そして、図6（e）に示すように移動完了時、制御ブロック430-2内の起点アドレスフィールド433および終点アドレスフィールド434に移動後のゲスト主記憶領域420-2の起点アドレスおよび終点アドレスを設定することにより、ゲスト主記憶装置420-2のアドレス変換情報を再設定する（ステップ5100）。次に、図6（f）に示すように、仮想計算機VM-2の制御ブロック430-2内のページ固定VM識別ビット432に1を設定し、ページ固定VMとする（ステップ5110）。また、仮想計算機VM-2上のOSによる非同期処理装置200の起動開始命令を直接実行するために、仮想計算機VM-2の制御ブロック430-2内の直接実行ビット431を1に設定する（ステップ5120）。

【0050】そして、図3の仮想計算機VM-2の制御ブロック430-2を指定してOSをディスパッチする（ステップ5130）。この後、OSの走行（ステップ5140）が開始される。このOSが走行中は、ゲスト絶対アドレスからホスト絶対アドレスへの変換は、仮想

計算機VM-2の制御ブロック430-2内のページ固定VM識別ビット432は1であるから、ゲスト絶対アドレスに起点アドレスフィールド433の値を加算することにより行なわれる。さらに、終点アドレスフィールド434の値と、ホスト絶対アドレスを比較することによって、アドレスの上限チェックが、行なわれる。

【0051】走行開始後、仮想計算機VM-2上のOSが非同期処理装置200の起動開始命令を発行すると（ステップ5150）、VM-2の制御ブロック430-2の直接実行ビット431は1なので、この命令は、仮想計算機モニタ410に割り出されことなく、命令処理装置100によって、直接実行される（ステップ5160）。

【0052】以上のようにして、本実施例によれば、中央処理装置とは非同期に処理を行なうチャネル等の非同期処理装置200によるゲスト主記憶領域へのアクセスを停止することなく、ゲスト主記憶領域を主記憶装置上で移動可能な仮想計算機システムを提供できる。また、動的に、ページ固定VMをページ可能VMに変更し、あるいは、ページ可能VMをページ固定VMに変更することができる。

【0053】II. 計算機資源の割り付け変更方法

図7、図8、および、図9を用いて計算機資源の割り付け変更方法を説明する。図7は計算機資源の割り付け変更方法のフローチャートであり、図8はゲスト主記憶領域のサイズの変更方法の説明図であり、図9は構成テーブル9000の説明図である。構成テーブル9000は、イベント発生前とイベント発生後において各仮想計算機に割り付ける計算機資源を示す。例えば、図9の構成テーブル9000は、図8（a）に示す初期状態（イベント発生前の状態）において、32メガバイトからなるゲスト主記憶領域420-1を有する仮想計算機VM-1と、128メガバイトからなるゲスト主記憶領域420-2を有する仮想計算機VM-2と、32メガバイトからなるゲスト主記憶領域420-1を有する仮想計算機VM-3の3台の仮想計算機が、中央処理装置100-1および100-2を共用して走行していることを表わしている。

【0054】この後、特定のイベントが発生すると仮想計算機モニタ410は、仮想計算機VM-2上のOSにゲスト主記憶領域420-2を構成テーブルに指定された32メガバイトに縮小するよう要求する（ステップ7000）。ここで、特定のイベントとは、特定の時間になったというイベントや、図1に示す第1の計算機500に障害が発生し、システム切り替え装置700によって、この第1の計算機500の処理を第2の計算機600の仮想計算機VM-1上のOSが引き継いだというイベントを示す。この結果、仮想計算機VM-2上のOSは、図8（b）に示すように、ゲスト主記憶領域420-2内の高位アドレスの領域をオフライン（未接続状

態)にし(ステップ7010)、仮想計算機VM-2から切り離す。

【0055】次に、仮想計算機モニタ410は、仮想計算機VM-2のゲスト主記憶領域420-2を、第1の実施例の場合と同様に、移動する(ステップ7020)。これによって、図8の(c)に示すように、仮想計算機VM-1のゲスト主記憶領域420-1に隣接した高位アドレスの領域を空き領域とすることができる。

【0056】また、仮想計算機モニタ410が、仮想計算機VM-1上のOSにゲスト主記憶領域420-1を構成テーブルに指定された128メガバイトに拡大するよう要求する(ステップ7030)。この結果、図8の(d)に示すように、仮想計算機VM-1上のOSが、ゲスト主記憶領域420-1に隣接した高位アドレスの領域をオンライン(接続状態)にし(ステップ7040)、仮想計算機VM-1に接続する。

【0057】さらに、仮想計算機モニタ410は、構成テーブル9000が示すように、1台の中央処理装置100-1を仮想計算機VM-1に専有させ、中央処理装置100-2を仮想計算機VM-2と仮想計算機VM-3で共有させて、仮想計算機上のOSを走行させる(ステップ7050)。

【0058】以上のようにして、本実施例によれば、ゲスト主記憶領域のサイズを動的に変更可能な仮想計算機システムを提供できる。また、仮想計算機への中央処理装置の割り付けを動的に変更可能な仮想計算機システムを提供できる。

【0059】

【発明の効果】本発明によれば、各仮想計算機のゲスト主記憶領域を主記憶装置上で移動して特定の仮想計算機のゲスト主記憶領域に隣接するアドレス高位の領域を空き領域にすることにより、ゲスト主記憶領域のサイズを動的に変更可能な仮想計算機システムを提供できる。

【0060】また、チャネル等の非同期処理装置によるゲスト主記憶領域のアクセス対象外の領域から順次、領域を移動することにより、非同期処理装置によるゲスト主記憶領域へのアクセスを停止することなく、ゲスト主記憶領域を主記憶装置上で移動可能な仮想計算機シス

テムを提供できる。

【0061】また、ページ固定VMのゲスト主記憶領域とこれが常駐する主記憶装置上の連続領域の対応関係をホスト仮想空間に定義することにより、ページ固定VMをページ可能VMに変更可能な仮想計算機システムを提供できる。

【0062】また、ホスト仮想空間のマッピング先として、主記憶領域の連続領域を対応付けることにより、ページ可能VMをページ固定VMに変更可能な仮想計算機システムを提供できる。

【0063】さらに、イベントの発生を契機として各仮想計算機に構成テーブルに記載された計算機資源を割り付けることにより、仮想計算機への中央処理装置の割り付けを動的に変更可能な仮想計算機システムを提供できる。

【図面の簡単な説明】

【図1】本発明の適用対象を例示するためのホストシステムの説明図。

【図2】本発明が適用される仮想計算機システムの説明図。

【図3】図2の仮想計算機の制御ブロックの説明図。

【図4】図2の仮想計算機システムにおけるアドレス変換の説明図。

【図5】本発明の第1の実施例におけるゲスト主記憶領域の移動方法のフローチャート。

【図6】本発明の第1の実施例におけるゲスト主記憶領域の移動方法の説明図。

【図7】本発明の第2の実施例における仮想計算機に対する計算機資源の割付け変更方法のフローチャート。

【図8】本発明の第2の実施例におけるゲスト主記憶領域のサイズの変更方法の説明図。

【図9】本発明の第2の実施例における構成テーブルの説明図。

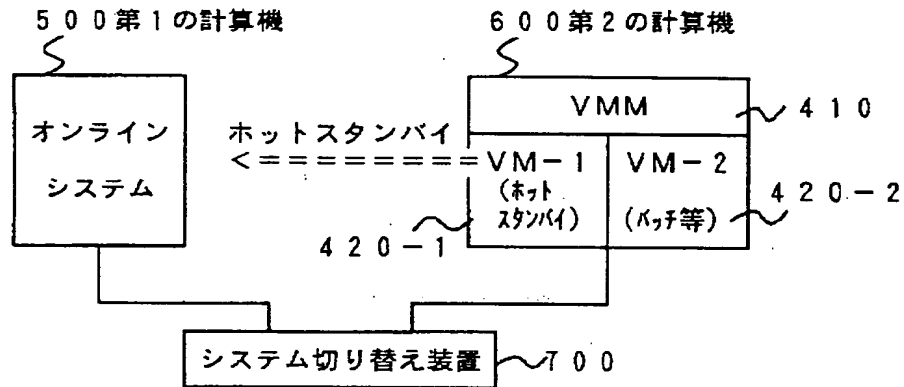
【符号の説明】

100：中央処理装置、200：非同期処理装置、300：記憶制御装置、400：主記憶装置、410：仮想計算機モニタ、420-i：仮想計算機VM-iのゲスト主記憶領域、9000：構成テーブル

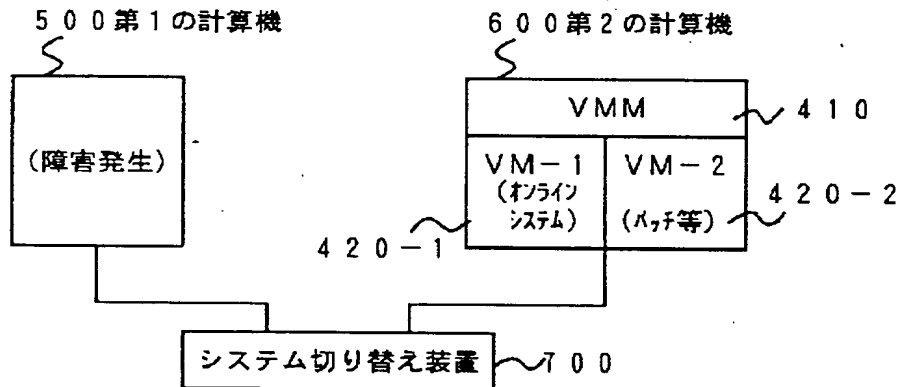
【図1】

ホットスタンバイシステムの説明図（図1）

(a) システム切り替え前

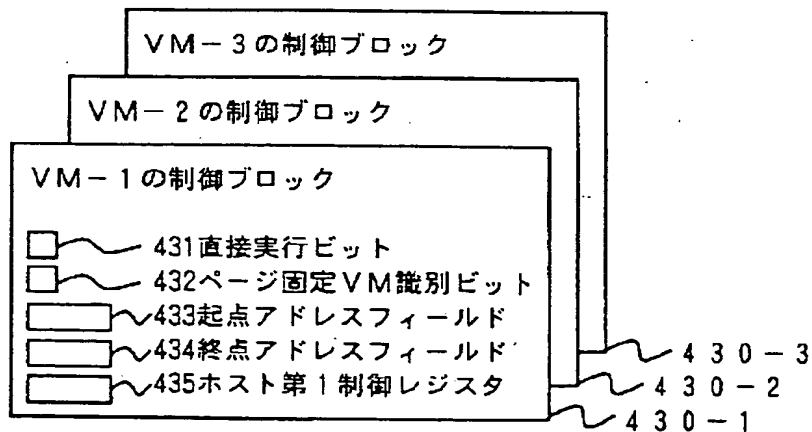


(b) システム切り替え後



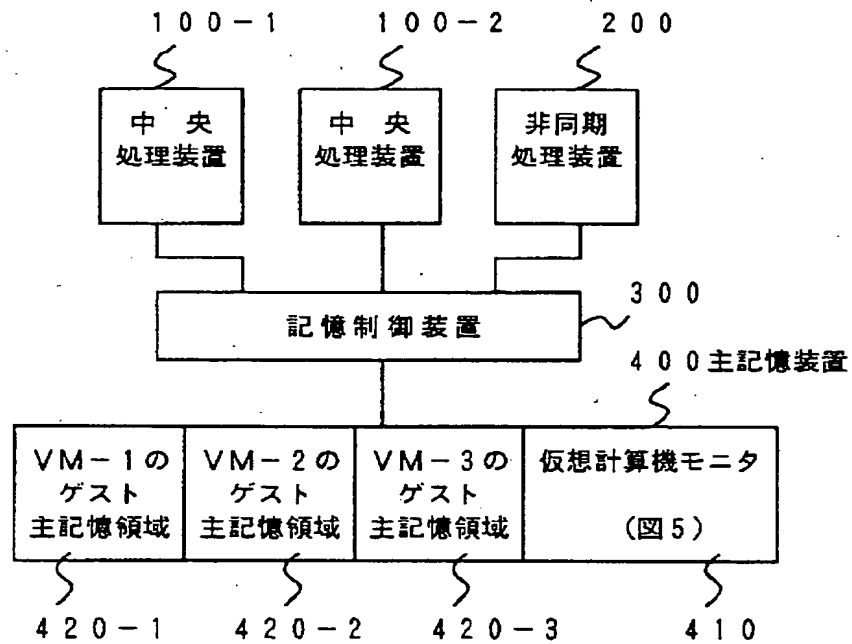
【図3】

仮想計算機（VM）の制御ブロックの説明図（図3）



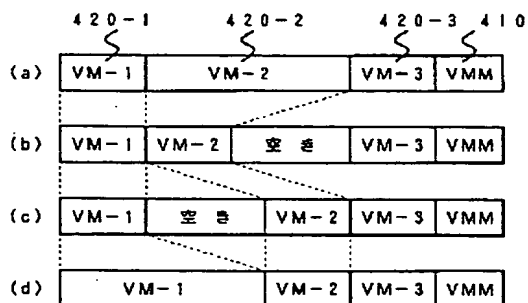
【図2】

仮想計算機システムの説明図（図2）



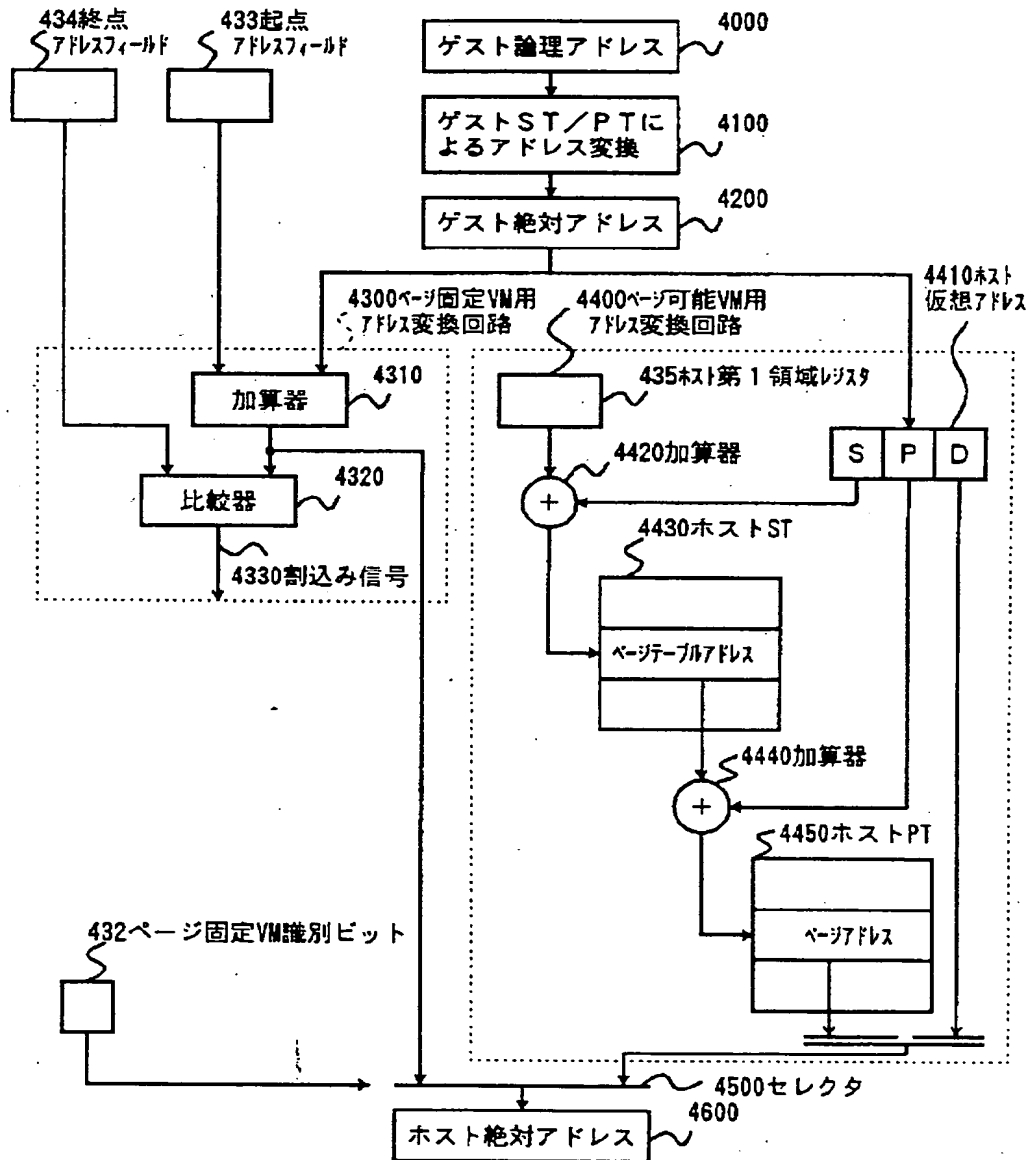
【図8】

ゲスト主記憶領域のサイズの変更方法の説明図（図8）

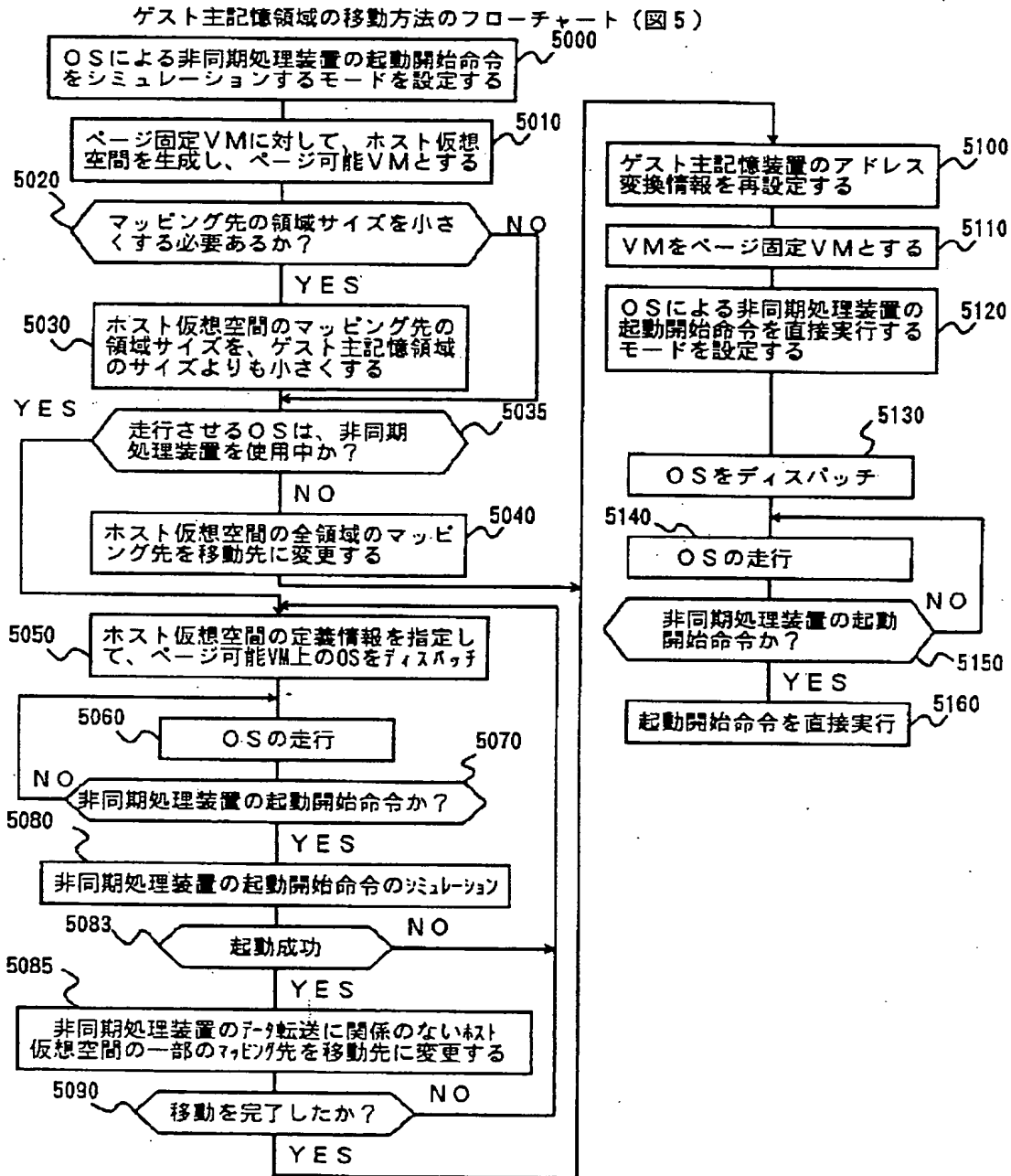


【図4】

仮想計算機システムにおけるアドレス変換の説明図（図4）

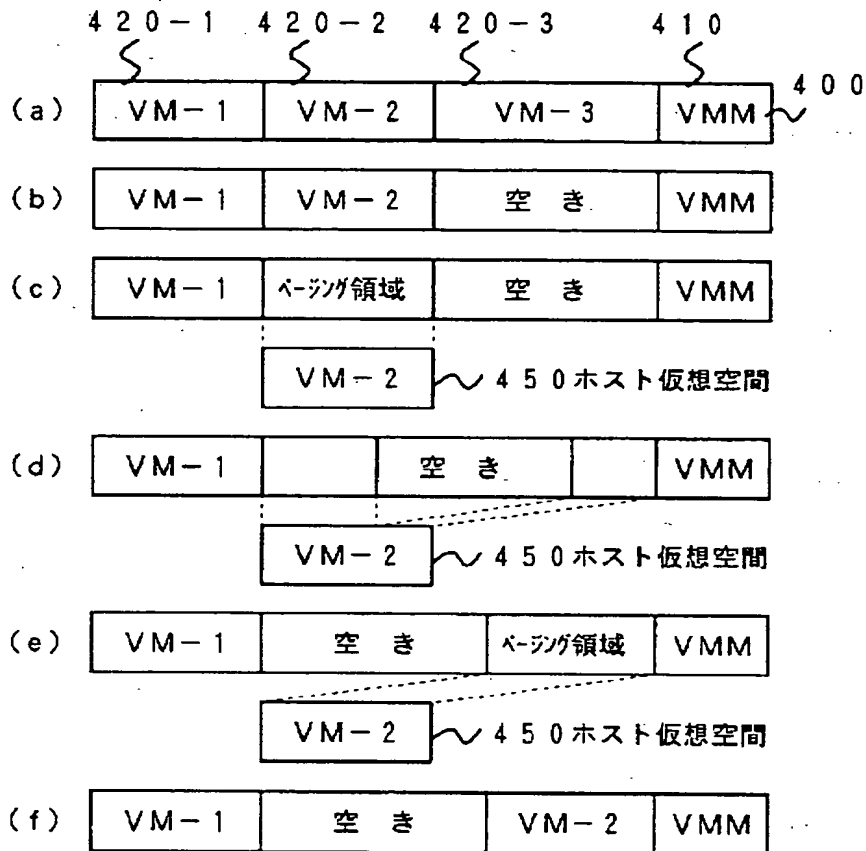


【図5】



【図6】

ゲスト主記憶領域の移動方法の説明図（図6）



【図9】

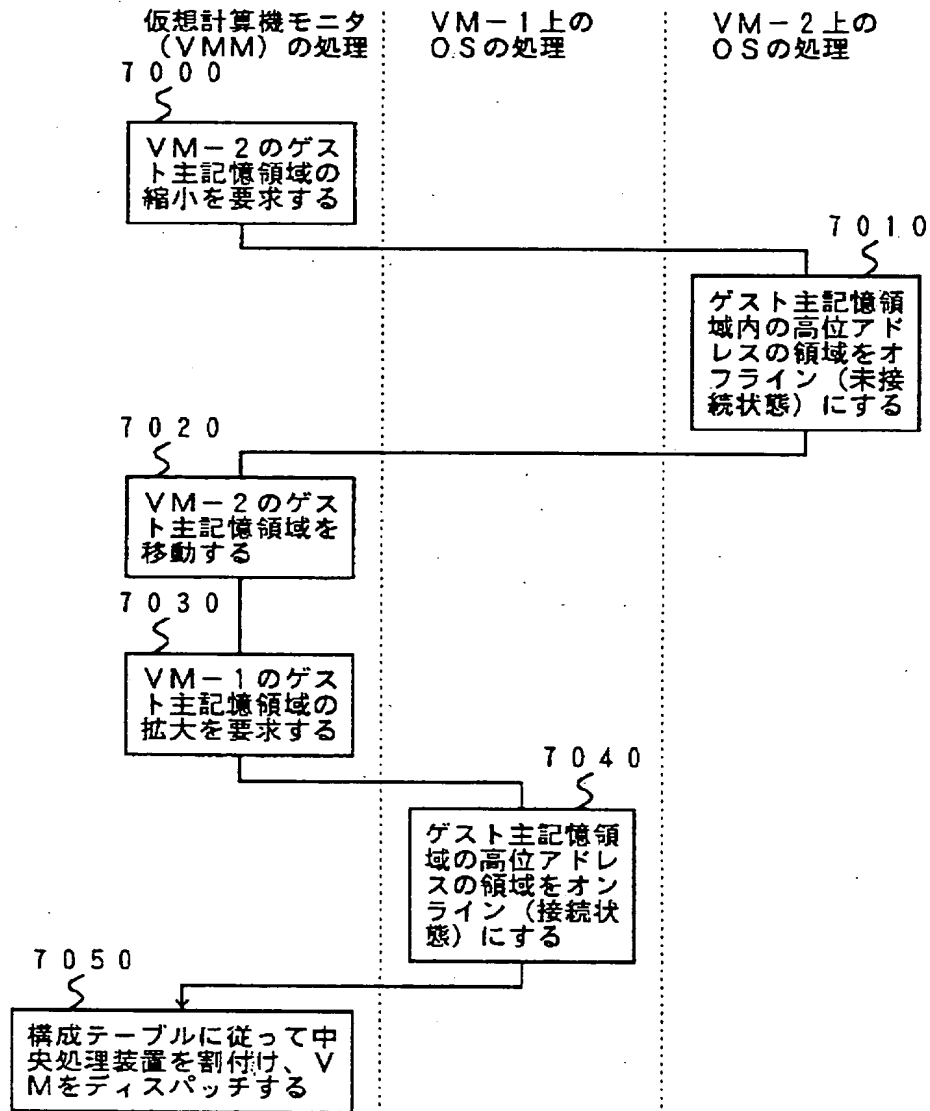
構成テーブルの説明図（図9）

9000 構成テーブル

		VM-1	VM-2	VM-3
イベント発生前	ゲスト主記憶サイズ	32MB	128MB	32MB
	中央処理装置の割り当て	共用	共用	共用
イベント発生後	ゲスト主記憶サイズ	128MB	32MB	32MB
	中央処理装置の割り当て	専有	共用	共用

【図7】

計算機資源の割り当て変更方法のフローチャート（図7）



フロントページの続き

(72)発明者 山本 正
 神奈川県横浜市戸塚区戸塚町5030番地 株
 式会社日立製作所ソフトウェア開発本部内

(72)発明者 池ヶ谷 直子
 神奈川県川崎市麻生区王禅寺1099 株式会
 社日立製作所システム開発研究所内